

מערכות תוכנה

הנדסאים – הנדסת אלקטרוניקה

הנחיות לנבחן

- א. משך הבחינה: ארבע שעות.
- ב. מבנה השאלון ומפתח ההערכה: בשאלון זה עשר שאלות. עליך לענות על חמש שאלות בלבד בהתאם לפירוט שלהלן.
- חלק א': 40 נקודות**
- שאלות 1-2. יש לענות על שאלה אחת. ערך כל שאלה – 20 נקודות.
- שאלות 3-4. יש לענות על שאלה אחת. ערך כל שאלה – 20 נקודות.
- חלק ב': 60 נקודות**
- שאלות 5-10. יש לענות על שלוש שאלות. ערך כל שאלה – 20 נקודות.
- בסך-הכול: 100 נקודות.**
- ג. חומר עזר מותר לשימוש:
- מחשבון – אין להשתמש במחשב כף יד או במחשבון המאפשר תקשורת חיצונית.
 - קלסר אחד בלבד עם חומר ההרצאות. אין להוציא דפים מהקלסר.
 - שני ספרי לימוד.
- ד. הוראות מיוחדות:
- יש לציין את המקור ואת מספר העמוד במקרים שבהם ניתנה תשובה מתוך ספרי הלימוד.
- ה. הוראות כלליות:
- יש לקרוא בעיון את ההנחיות בדף השער ואת כל שאלות הבחינה ולוודא שהן מובנות.
 - יש להשאיר את העמוד הראשון במחברת הבחינה ריק. בסיום המבחן יש לרשום בעמוד זה את מספרי התשובות לבדיקה. התשובות ייבדקו לפי סדר כתיבתן בעמוד זה. לא ייבדקו תשובות עודפות.
 - יש לכתוב את התשובות במחברת הבחינה בעט בלבד, בכתב יד ברור.
 - יש להתחיל כל תשובה בעמוד חדש ולציין את מספר השאלה ואת הסעיף. אין צורך להעתיק את השאלה עצמה.
 - טיוטה יש לכתוב במחברת הבחינה בלבד. יש לרשום את המילה "טיוטה" בראש העמוד ולהעביר עליו קו כדי שלא ייבדק.
 - יש להציג פתרון מלא ומנומק כולל חישובים לפי הצורך. הצגת תשובה סופית ללא שלבי הפתרון לא תזכה בניקוד.
 - אם לדעתך חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לך להמשיך בפתרון השאלה. נמק את בחירתך.

חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה!
ההנחיות בשאלון זה מנוסחות בלשון זכר, אך מכוונות לנבחנות ולנבחנים כאחד.

חלק א' – תוכנה C++ (40 נקודות)

ענה על אחת מבין השאלות 1-2 (לכל שאלה – 20 נקודות).

שאלה 1**הגדרות**

- בהינתן מטריצה M מסדר $rows \times SIZE$, כאשר $rows$ ו- $SIZE$ שלמים חיוביים, ובהינתן שלם חיובי $partSize$, נגדיר **כמטריצת בסיס** את תת-המטריצה הריבועית בגודל $partSize$ שהפינה השמאלית העליונה שלה היא הפינה השמאלית העליונה של המטריצה M .
למשל, במטריצה הבאה כאשר $partSize=2$ מטריצת הבסיס היא המטריצה שהפינה השמאלית העליונה שלה נמצאת במיקום $[0,0]$ והפינה הימנית התחתונה במיקום $[1,1]$ במטריצה M (האיברים שתוכנם מסומן ב-X):

	0	1	2	3	4	5	6	7
0	X	X						
1	X	X						
2								
3								
4								
5								

- מטריצה חופפת למטריצת הבסיס במטריצה M היא מטריצה שמיקום הפינה השמאלית העליונה שלה $[r,c]$ מקיים: $rows \% r = 0$ וגם $SIZE \% c = 0$. כך מתקבלת תת-מטריצה נוספת במטריצה M שהאינדקסים שלה הם בהתאמה למטריצת הבסיס.
- ניתן להניח שאורך ורוחב המטריצה מתחלקים ב- $partSize$ ללא שארית.
למשל, במטריצה M חופפות למטריצת הבסיס מטריצות ריבועיות בגודל $partSize$ שהפינה השמאלית העליונה שלהן נמצאת במיקומים:
 $[0,2]$ $[0,4]$ $[0,6]$ $[2,0]$ $[2,2]$ $[2,4]$ $[2,6]$ $[4,0]$ $[4,2]$ $[4,4]$ ו- $[4,6]$

כתוב פונקציה לפי הדרישות הבאות:

- הפונקציה מקבלת מספר חיובי שלם המייצג את מספר השורות במטריצה, מספר חיובי שלם המייצג את מספר העמודות במטריצה ומספר חיובי שלם עבור ממד המטריצה הבסיסית.
- עבור מטריצה המיוצגת על-ידי הממדים שהועברו, הפונקציה מדפיסה עבור כל איבר במטריצת הבסיס את כל האינדקסים התואמים של כל המטריצות החופפות שגודלן זהה לגודל מטריצת הבסיס.

הערות:

- ניתן להניח ש: $partSize < SIZE$, $partSize < rows$ ומתחלקים ב- $partSize$ ללא שארית.
- פורמט ההדפסה – ראה דוגמאות להלן.

דוגמאות:

- עבור המטריצה הנ"ל עם 6 שורות ו-8 עמודות כאשר $partSize=2$, הפונקציה תדפיס:

[0,0]	→	[2,0]	[4,0]	[0,2]	[2,2]	[4,2]	[0,4]	[2,4]	[4,4]	[0,6]	[2,6]	[4,6]
[0,1]	→	[2,1]	[4,1]	[0,3]	[2,3]	[4,3]	[0,5]	[2,5]	[4,5]	[0,7]	[2,7]	[4,7]
[1,0]	→	[3,0]	[5,0]	[1,2]	[3,2]	[5,2]	[1,4]	[3,4]	[5,4]	[1,6]	[3,6]	[5,6]
[1,1]	→	[3,1]	[5,1]	[1,3]	[3,3]	[5,3]	[1,5]	[3,5]	[5,5]	[1,7]	[3,7]	[5,7]

- עבור המטריצה הבאה עם 6 שורות ו-9 עמודות כאשר $partSize=3$:

	0	1	2	3	4	5	6	7	8
0									
1									
2									
3									
4									
5									

הפונקציה תדפיס:

[0,0]	→	[3,0]	[0,3]	[3,3]	[0,6]	[3,6]
[0,1]	→	[3,1]	[0,4]	[3,4]	[0,7]	[3,7]
[0,2]	→	[3,2]	[0,5]	[3,5]	[0,8]	[3,8]
[1,0]	→	[4,0]	[1,3]	[4,3]	[1,6]	[4,6]
[1,1]	→	[4,1]	[1,4]	[4,4]	[1,7]	[4,7]
[1,2]	→	[4,2]	[1,5]	[4,5]	[1,8]	[4,8]
[2,0]	→	[5,0]	[2,3]	[5,3]	[2,6]	[5,6]
[2,1]	→	[5,1]	[2,4]	[5,4]	[2,7]	[5,7]
[2,2]	→	[5,2]	[2,5]	[5,5]	[2,8]	[5,8]

שאלה 2

כתוב פונקציה **רקורסיבית** המקבלת שני מערכים באורך שווה ואת אורכם. הפונקציה תחזיר true אם כל הערכים במקומות הזוגיים במערכים שווים, אחרת הפונקציה תחזיר false.

חתימת הפונקציה:

```
bool equalOnEvenLocations(int arr1[], int arr2[], int size)
```

דוגמאות:

- עבור המערך {2,5,8,2,6,9} והמערך {2,8,8,3,6,7} שאורכם 6 יוחזר true מאחר שבמיקומים 0, 2 ו-4 הערכים בשני המערכים זהים.
- עבור המערך {2,5,7,2,6,9} והמערך {2,8,8,3,6,7} שאורכם 6 יוחזר false מאחר שבמיקום ה-2 הערכים בשני המערכים שונים.
- עבור מערכים עם 0 איברים יוחזר true.

ענה על אחת מבין השאלות 3-4 (לכל שאלה – 20 נקודות).**שאלה 3**

להלן קוד המגדיר מחלקה ופונקציה ראשית. כתוב מה יהיה פלט התוכנית.

```
#include <iostream>
using namespace std;
#include <string.h>
class A
{
public:
    A() {cout << "In A::A\n";}
    A(const A& other) {cout << "In A::A(copy)\n";}

    const A& operator=(const A& other)
    {
        cout << "In A::A(copy)\n";
        return *this;
    }
};

class B
{
    A a1;
public:
    B() {cout << "In B::B\n";}
    B(const B& other) {cout << "In B::B(copy)\n";}
    ~B() {cout << "In B::~~B\n";}
};

class C : public B
{
    A a2;
public:
    C() {cout << "In C::C\n";}
    C(const A& a, const B& b) : B(b), a2(a)
    {
        cout << "In C::C(A, B)\n";
    }
};

void main()
{
    A a1;
    cout << "1 ----- \n";
    A a2(a1);
    cout << "2 ----- \n";
    B b1;
    cout << "3 ----- \n";
    B b2(b1);
    cout << "4 ----- \n";
    C c1(a1, B());
    cout << "5 ----- \n";
    C c2;
    cout << "6 ----- \n";
    c2 = c1;
    cout << "7 ----- \n";
}
```

שאלה 4

להלן קוד המגדיר מחלקה ופונקציה ראשית. כתוב מה יהיה פלט התוכנית.

```
#include <iostream>
using namespace std;

class Stam
{
    char* str;
    int x;
    double* d;

public:
    Stam(char* str, int x, double* d) : str(str), x(x), d(d) {}

    char* getStr() {return str;}
    int getX() {return x;}
    double* getD() {return d;}

    ~Stam()
    {
        *d *= 2;
        str[x] = '*';
    }
};

void foo(double d[], char text[])
{
    int n[3] = {3,2,1};

    Stam s1(text, n[1], &d[2]);
    Stam s2(text+s1.getX(), (int)d[1], &d[0]);
    Stam s3(text+s2.getX()+1, 8, s1.getD());

    Stam* arr[3] = {&s1, &s2, &s3};
    for (int i=1 ; i < 3 ; i++)
    {
        cout << arr[i]->getX() << " ";
        cout << *arr[i]->getD() << " ";
        cout << arr[i]->getStr()+arr[i-1]->getX() << "\n";
    }

    cout << text << endl;
}

void main()
{
    double d[3] = {1.1, 2.2, 3.3};
    char text[] = "abcdefghijkl";
    foo(d, text);

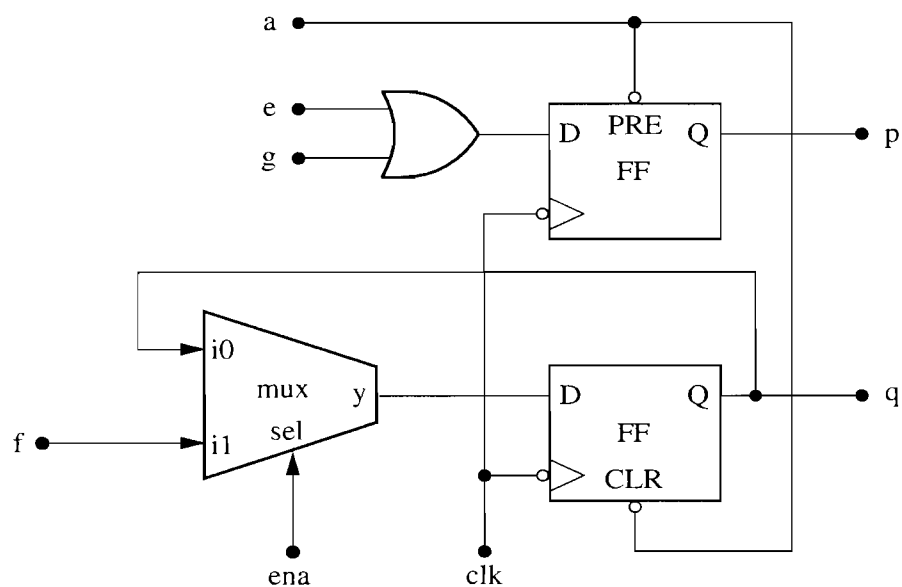
    for (int i=0 ; i < 3 ; i++)
        cout << d[i] << " ";
    cout << endl << text << endl;
}
```

חלק ב' – VHDL (60 נקודות)

ענה על שלוש מבין השאלות 5-10 (לכל שאלה – 20 נקודות).

שאלה 5

באיור לשאלה 5 נתון תיאור סכמטי של מערכת חומרה:



איור לשאלה 5

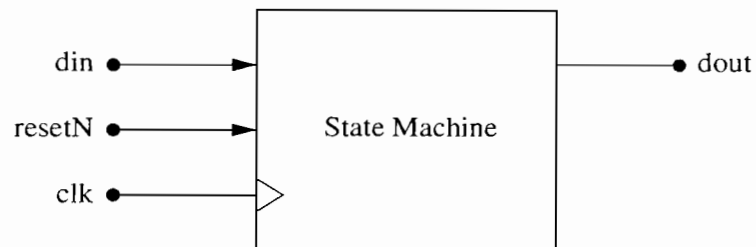
כתוב את הקוד (ישות וארכיטקטורה מלאים) המתאר את המערכת בשפת VHDL. השתמש בתהליך סינכרוני אחד עבור תיאור של כל המערכת.

הנחיה:

השתמש רק באותות המסומנים באיור לשאלה 5 ללא תוספת של אותות כלשהם. אין להשתמש בשמות האותות הרשומים בתוך הרכיבים.

שאלה 6

באיור לשאלה 6 מתוארת מכונת מצבים בעלת שלושה מבואים ומוצא יחיד.



איור לשאלה 6

נתונה תכנית VHDL המתארת את המכונה:

```

library ieee ;
use ieee.std_logic_1164.all ;
entity sm is
  port ( resetN , clk, din, restart : in  std_logic ;
         dout                : buffer std_logic ) ;
end sm ;
architecture arc_sm of sm is
  type state is (pre,bingo,post) ;
  signal present_state , next_state : state ;
begin
  process ( resetN , clk )
  begin
    if resetN = '0' then
      present_state <= pre ;
    elsif clk'event and clk = '1' then
      present_state <= next_state ;
    end if ;
  end process ;
  process ( present_state , din , restart )
  begin
    dout <= '0' ;
    case present_state is
      when pre =>
        if din = '1' then
          next_state <= bingo ;
        else
          next_state <= pre ;
        end if ;
      when bingo =>
        dout <= '1' ;
        next_state <= post ;
      when post =>
        if restart = '1' then
          next_state <= pre ;
        else
          next_state <= post ;
        end if ;
    end case ;
  end process ;
end arc_sm ;
  
```

```
when others      =>
    next_state <= pre ;
end case;
end process ;
end arc_sm ;
```

(10 נק') א. סרטט דיאגרמת מצבים (דיאגרמת בועות) של המכונה.

(10 נק') ב. כתוב מחדש את הקוד של המכונה כאשר אתה משתמש באות סימבולי שנקרא -

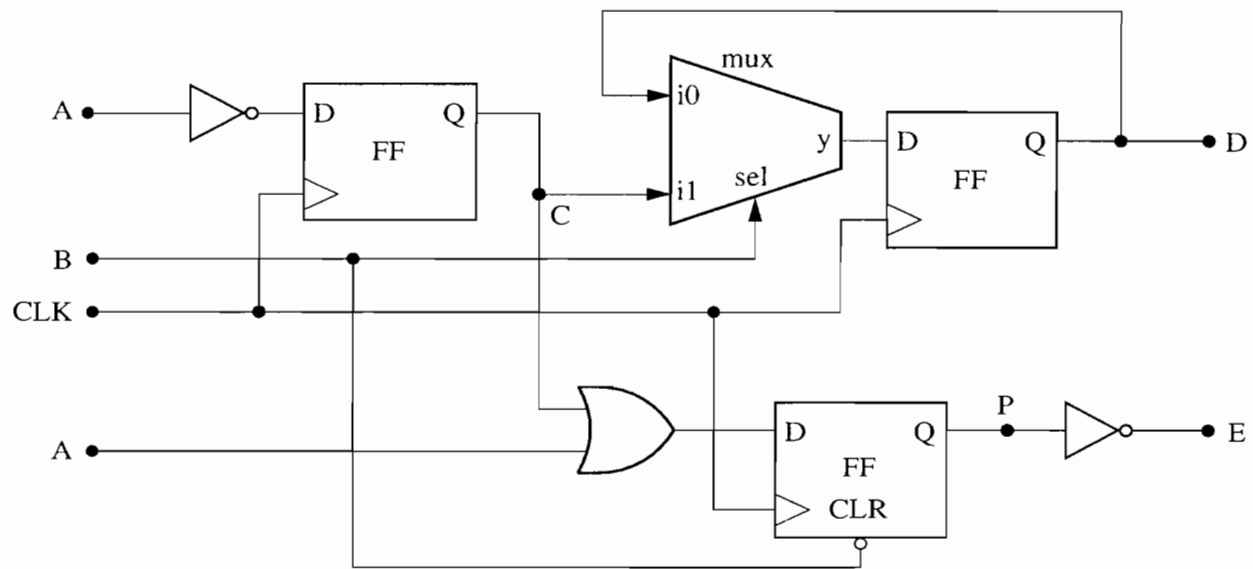
Present State וללא שימוש באות - Next State (One Segment Coding) .

הנחיות:

1. הקוד צריך לכלול תהליך סינכרוני שמתאר הן את רגיסטר המצב והן את לוגיקת העירור למצב הבא של המכונה.
2. תאר את לוגיקת המוצא בחלק נפרד בארכיטקטורה (כלומר לא בתהליך הנ"ל).
3. אין צורך לכתוב שוב את הישות.

שאלה 7

באיור לשאלה 7 נתון תיאור סכמטי של מערכת חומרה:



איור לשאלה 7

כתוב את הקוד (ישות וארכיטקטורה מלאים) המתאר את החומרה בשפת VHDL. בכתיבת הקוד השתמש בשני תהליכים סינכרוניים (חשוב להפריד בין שני סוגי החלקים הסינכרוניים).

הנחיות:

1. מותר להשתמש בהשמה צירופית.
2. אסור להשתמש במשתנים או באותות נוספים פרט לאלו שמצוינים באיור לשאלה 7.
3. אסור להשתמש בשמות האותות שרשומים בתוך הרכיבים.

שאלה 8

נתונה מערכת לזיהוי הקוד "1110", כמתואר באיור לשאלה 8.

במבוא הטורי של המערכת SI מתקבלת סיבית אחת בכל דרבון קצה חיובי של המבוא CLK.



איור לשאלה 8

המוצא Z הופך מ'0' ל'1' רק בסיום קליטת רצף סיביות הקוד "1110". לצורך זיהוי הקוד, הסיבית האחרונה יכולה להיחשב כסיבית הראשונה של הרצף הבא, כפי שמוצג בדוגמה הבאה:

כניסת SI	1	1	1	0	1	1	1	1	1	0
מוצא Z	0	0	0	1	0	0	0	0	0	1

א. (10 נק') סרטט דיאגרמת בועות המתארת את פעולת המערכת וכוללת גם ציון העירור על החצים בדיאגרמה.

ב. (10 נק') כתוב תכנית מלאה בשפת VHDL המבצעת את פעולת המערכת המזהה את הקוד "1110".

שאלה 9

בשאלה זו נתונים קטעי קוד בשפת VHDL. כל קטע קוד מתאר מימוש של רכיב חומרה מסוים. עליך לסרטט את רכיב החומרה המתאים לכל קטע קוד.

כל האותות ברכיבים הם סיביות בודדות (כלומר סקלריות) מסוג std_logic או bit (בחר לפי הצורך). הרכיבים שעומדים לרשותך לצורך ביצוע המימוש הם: שערים לוגיים, בוררי פליפ-פלופ, נועלים עם מבוא איפשור (Gated Latch), מעגלים תלת-מצביים (Tri-State), נגדים, מתחי ספק וחיווט חשמלי.

הערה: לצורך מימוש של רכיב חומרה אין צורך להשתמש בכל הרכיבים הנ"ל.

(10 נק') א.

```
library ieee ;
use ieee.std_logic_1164.all ;
entity mderiv is
    port ( resetN , clk : in  std_logic ;
          din           : in  std_logic ;
          qr,qf,qrs,qfs : buffer std_logic ) ;
end mderiv ;
architecture arc_mderiv of mderiv is
    signal x1 , x2 : std_logic ;
begin
    process ( resetN , clk )
    begin
        if resetN = '0' then
            x1 <= '0' ; x2 <= '0' ;
            qrs <= '0' ; qfs <= '0' ;
        elsif clk'event and clk = '1' then
            x1 <= din ; x2 <= x1 ;
            qrs <= x1 and not x2 ;
            qfs <= not x1 and x2 ;
        end if ;
    end process ;
    qr <= x1 and not x2 ;
    qf <= not x1 and x2 ;
end arc_mderiv ;
```

(10 נק') ב.

```
library ieee ;
use ieee.std_logic_1164.all ;
entity trans is
    port ( a , b : inout std_logic_vector (3 downto 0) ;
          enable : in  std_logic ) ;
end trans ;
architecture arc_trans of trans is
begin
    a <= "ZZZZ" when enable = '0' else b ;
    b <= "HHHH" when enable = '1' else a ;
end arc_trans ;
```

שאלה 10

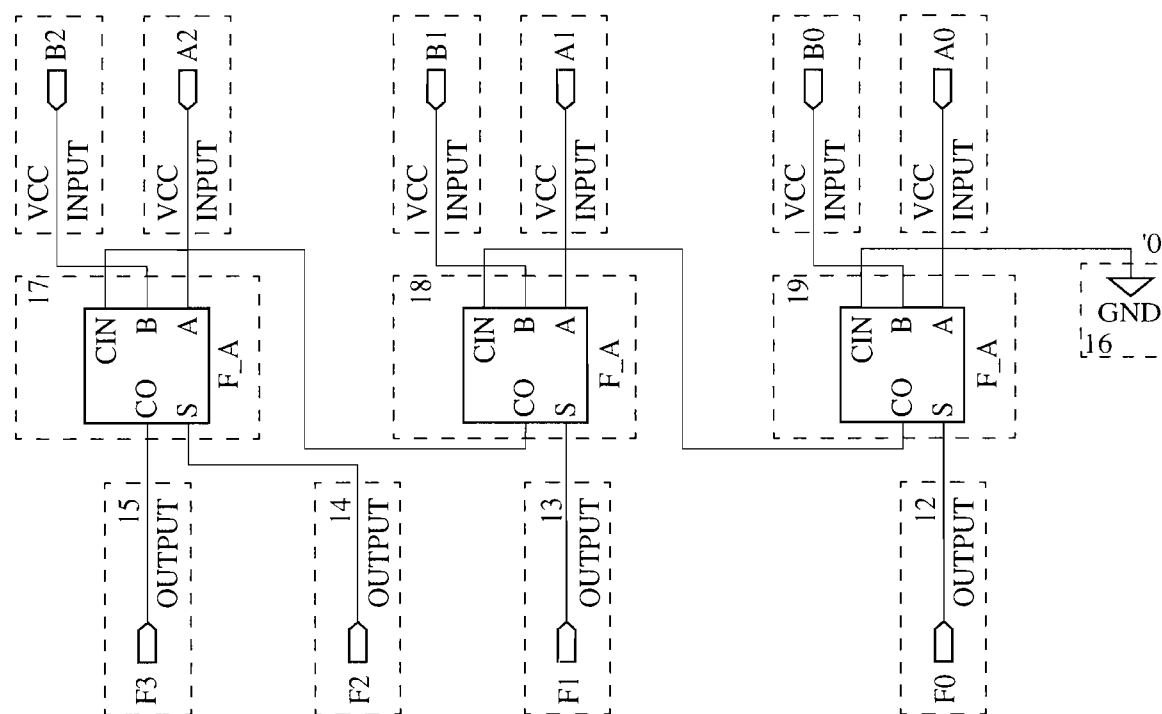
10 נק') א. כתוב תכנית מלאה בשפת VHDL המבצעת פעולת חיבור של שלוש סיביות ומציגה את תוצאת החיבור במוצא המערכת (מחבר מלא – FULL ADDER).

על התכנית לכלול את היישות F_A (ENTITY) הבאה:

```
entity F_A is
port ( A,B,Cin : in bit ;
      S,Co : out bit ) ;
end;
```

10 נק') ב. באיור לשאלה 10 ב' מתוארת מערכת המבצעת פעולת חיבור בין שני מספרים (B, A) בעלי שלוש סיביות כל אחד. המימוש נעשה באמצעות שרשר של שלושה מחברים מלאים (FULL ADDERS).

כתוב תכנית מלאה בשפת VHDL המבצעת חיבור בשיטת התכנות המבני לתיאור החיבור בין מרכיבי המערכת (COMPONENTS).



איור לשאלה 10 ב'

הנחיות:

- יש להשתמש בפקודת PORT MAP לצורך החיבור בין מרכיבי המערכת.
- השימוש בפקודה GENERATE מומלץ אך אינו חובה.
- יש להעתיק את סרטוט המערכת כדי להציג את המיקומים והשמות לסיגנלים שבחרת לצורך החיבור.
- הבא בחשבון שקטע התכנית של כל מחבר F_A כתוב, מהודר ומכיל את היישות הנתונה בחלק א' של השאלה.

בהצלחה!